Remarks

Status of application

Claims 1-7, 9-21, and 23-40 are pending and stand rejected in view of prior art. Further to the Remarks made below, reexamination and reconsideration are respectfully requested.

The invention

For a brief description of Applicant's invention, please refer to the previously-filed Amendment, which was filed on 01/02/2007.

Prior art rejections

Section 103 rejection: Lomet, Lahey, Klotz, Schmidt, and Klemm

Claims 1-7, 9-21, and 23-40 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Lomet (Patent Number 6,490,594) in view of Lahey et al. ('Lahey' hereinafter) (Patent Number 7,028,303) and further in view of Klotz et al. ('Klotz' hereinafter) (Publication Number 2004/0015762) and further in view of Schmidt et al. ('Schmidt' hereinafter) ('Alleviating Priority Inversion. and Non-determinism in Real-time CORBA ORB Core Architectures", 4th IEEE Real-time Technology and Applications Symposium, Denver, CO, June 3-5, 1998) and further in view of Klemm et al. ('Klemm' hereinafter) (Patent Number 6,457,142).

In response to Applicant's previously filed Amendment and Request for Continued Examination, the Examiner withdrew the previous rejection based on the combination of Lomet, Lahey, Klotz, and Schmidt. However, the Examiner simply adds another reference to create what is now a very long chain of references (and logic), with the Examiner relying a combined total of five references -- a sizable collection. To be sure, there is no absolute cap or ceiling as to the number of references that may form a competent combination under Section 103, but the fact that the Examiner is having to go to so many places to pull together an "obviousness" rejection here begs the question what exactly is "obvious." At some point, the thread of logic used to weave together such a large number of references becomes stretched so thin that it breaks. In particular in the present application, it is respectfully submitted that the Examiner has combined together

such a disparate and large collection of art, including references that have nothing to do with one another other than generally pertaining to computer-related technology, that the rejection does not establish obviousness under Section 103.

Notwithstanding the Amendment and RCE, the Examiner still relies as the core part of the rejection on the combination of: Lomet which describes database recovery technique, Lahey and Schmidt which describe the spawning of threads, and Klotz which describes a test suite that writes test data to figure out I/O performance. The Examiner combines these references together with little regard as to how well the individual pieces (i.e., individual references) fit together. Applying an analogy here, it is as if the Examiner is contending that he has a bucket of bolts and spare parts that he can shake up to re-create any imaginable combination, such as a futuristic mode of transportation. Such logic, however, fails to take into account that the combination that is now so "obvious" to re-create is in fact being re-created with the hindsight benefit of someone else's knowledge.

Here, Applicant has invented a database restore technique that is dynamically self-tuning. Applicant does not claim to have invented the notion of database recovery technique, nor the notion of spawning threads up to a preselected maximum number, nor the notion of writing test data for figuring out/measuring I/O (input/output) performance. Applicant is the first one, however, to have invented a dynamically self-tuning database restore technique. Applicant's invention includes features that overlap with some elements from the prior art but, as shown below, Applicant's claimed invention combines elements in a non-obvious way to provide a self-tuning database restore technique, which tunes itself on-the-fly (i.e., during actual production use) using live or real data (not test data).

As previously described in Applicant's prior response, Applicant's invention measures I/O performance and uses that result as a determining factor as to whether to spawn additional threads or not. In this manner, Applicant's improved database restore methodology may dynamically adapt to the then-current environment for reaching maximum throughput. At this point in the rejection, the tenuous nature of the Examiner's logic becomes evident as he begins ascribing features to the prior art (really, lifted from Applicant's invention) that the prior art not only does not have, but directly contradicts

what the prior art teaches one to do.

Consider how Applicant's invention functions and how the prior art functions. In Applicant's invention, the improved database restore methodology dynamically adapts to the throughput achieved by the current environment, in order to determine what number of threads will be employed. In order to reproduce this aspect of Applicant's invention, the Examiner needs to modify the particular teaching of Lahey, so that instead of Lahey's static setting (of maximum number of threads) Lahey is somehow adapted to dynamically set maximum threads spawned. However, as the Examiner has already acknowledged, Lahey does not include any mechanism for monitoring throughput (during actual use) that would allow Lahey to dynamically adapt the maximum number of threads spawned. So the Examiner turns to Schmidt.

However, Schmidt does not help. The Examiner cites Schmidt for the proposition that it teaches dynamic tuning, but a review of Schmidt reveals that it has no such teaching. Instead, it simply repeats the approach of Lahey (above) to spawn a number of threads up to a maximum amount. Consider the following specific teaching from Schmidt (at 2.1.2, the section cited by the Examiner):

> In the thread-pool concurrency architecture, the application initially spawns a fixed number of threads. In addition, when the initial thread pool size is insufficient, miniCOOL can be configured to dynamically spawn threads on behalf of server applications to handle requests, up to a maximum limit.

Here, Schmidt is problematic on two fronts. First, he simply describes that additional threads are spawn in response to additional incoming requests. No consideration is given by Schmidt in this description whether such additional spawning of threads may degrade I/O performance (e.g., due to memory constraints). Second, Schmidt basically teaches the same approach of Lahey, that is, spawning threads up to some preselected static setting (i.e., maximum number of threads). Importantly, Schmidt -- unlike Lahey -- simply describes the increased usage of threads (i.e., spawning more threads). Neither Schmidt nor Lahey can be properly construed to teach the dynamic adjustment or self

tuning approach of Applicant's invention, which includes not only increasing the usage of threads based on dynamic reading of I/O performance but also decreasing the usage of threads (by putting threads to sleep) should degradation of I/O performance be detected in real time.

To the extent that the Examiner suggests that Klotz's I/O measurements could be combined with either Schmidt or Lahey to reproduce this aspect of Applicant's invention, that suggested combination certainly ignores the underlying teachings of each respective reference. Both Schmidt's and Lahey's maximum thread number is preselected (i.e., statically set), thus making it clear that those systems are not set up to receive any sort of input that would allow the maximum to be dynamically set at run time (i.e., during actual system use). Clearly, both Schmidt and Lahey include no description whatsoever pertaining to any sort of dynamic adjustment that can also decrease the usage of threads, for example by putting threads to sleep as required by Applicant's independent claims. If anything, both Schmidt and Lahey teach simple thread usage of spawning additional threads up to a statically set maximum -- an approach that teaches away from Applicant's dynamic adaptation approach.

Klotz, for its part, provides nothing that would support the Examiner's contention that the Klotz's measurements could be fed back to either Schmidt's or Lahey's system in a manner that would re-create Applicant's invention. Klotz provides a test suite that writes test data in order to measure I/O performance. The Examiner does not provide any explanation of how Klotz's measurements could be tied to Schmidt's or Lahey's maximum thread number. If Klotz's measurements were passed to those systems, one would still have a system (as taught by Lahey or Schmidt) that specifies a maximum thread number spawned based on a preselected or static number (e.g., preconfigured by the user, before it is needed at run time).

Moreover, Applicant's invention performs the self tuning claim limitation using the live or actual data (i.e., the real data from the production database). Measurements of I/O performance are taken as this live data is being processed (i.e., reading data from disk, for restoring the database). This is very different than using a test case or test suite writing test data, such as described by Klotz. If anything, Klotz teaches away from Applicant's approach of measuring I/O performance based on how well the actual

production data (i.e., the real data from the database) is being processed (e.g., read or written to disk) by the I/O subsystem as the restore process is actually running. Importantly, the computing environment in which Applicant's system is deployed (i.e., one or more server computers with network connectivity) is not some sort of static environment. Instead, the actual demands placed on available computing resources in that environment change from moment to moment. The resources available in the environment at any given moment are highly variable, and thus optimum throughput is best achieved using an approach that dynamically adapts at run time, when it is actually needed. This level of fine-tuning is not taught or suggested by the cited art.

To shore up the above base combination (i.e., first four references) of Lomet, Lahey, Klotz, and Schmidt, the Examiner now adds Klemm, which seems to have even less to do with database art than the object request broker ("ORB") art of Schmidt. The Examiner adds Klemm for the proposition that it teaches the dynamic adjustment aspect of Applicant's invention as Klemm discusses putting threads to sleep. Putting threads to sleep by itself is of course a known concept, and Applicant certainly makes no claim to have invented that. Turning to the actual teaching of Klemm cited by the Examiner (at column 4, lines 20-24): one finds it simply teaches enforcing a maximum number of threads:

> If more than 250 threads of class PrefetchURL are executing concurrently suspend all newly spawned threads. Notify remote manager if this happens more than 10 times within 3600000 ms.

Here, as was the case with Lahey and Schmidt, Klemm is simply specifying a statically set maximum number of threads (e.g., 250) to use; if the maximum is exceeded, the Klemm system suspends any new threads (so that the maximum is enforced). There is no teaching or suggestion in Klemm that the number of threads used is dynamically adjusted up or down across a significant range to reach a nonstatic optimized value -- that is, a dynamically determined number of threads that gives optimum performance. The contention that Klemm teaches the dynamic adjustment of threads used -- up or down -- ignores the teaching of Klemm itself. Klemm simply enforces a preselected maximum

number of threads (i.e., a very specific and static number), and suspends new threads once that maximum is reached.

In Applicant's prior responses, Applicant's independent claims have been amended to highlight these "dynamic adaptation" and "self-tuning" features of Applicant's invention. For the reasons stated above, it is believed that these features distinguished over the five-way combination of art references now relied upon by the Examiner. In view of the clarifying remarks made above as well as prior amendments made to the claims, it is respectfully submitted that the claims set forth a patentable advance over the art and that any rejection under Section 103 is overcome.

Any dependent claims not explicitly discussed are believed to be allowable by virtue of dependency from Applicant's independent claims, as discussed in detail above.

Conclusion

In view of the foregoing remarks and the amendment to the claims, it is believed that all claims are now in condition for allowance. Hence, it is respectfully requested that the application be passed to issue at an early date.

If for any reason the Examiner feels that a telephone conference would in any way expedite prosecution of the subject application, the Examiner is invited to telephone the undersigned at 408 884 1507.

Respectfully submitted,

Date: January 18, 2008

/John A. Smart/

John A. Smart; Reg. No. 34,929
Attorney of Record

408 884 1507
815 572 8299 FAX